

Class Alien

Namespace: [Model](#)

Assembly: Model.dll








Classe contenant toutes les informations sur les aliens du jeu ainsi que les actions qu'ils peuvent réaliser

```
public class Alien
```

Inheritance

[object](#)  ← Alien

Inherited Members

[object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Constructors

Alien(int, int)

Constructeur de la classe alien

```
public Alien(int alienX, int alienY)
```

Parameters

alienX [int](#) 

Position X de l'alien

alienY [int](#) 

Position Y de l'alien

Fields

SCREENWIDTH

```
public const int SCREENWIDTH = 153
```

Field Value

[int](#)

alienDead

Variable qui dis si l'alien est considéré comme mort ou vivant

```
public bool alienDead
```

Field Value

[bool](#)

alienRight

Variable qui dis si l'alien peut aller à droite

```
public bool alienRight
```

Field Value

[bool](#)

deadDrawPassed

```
public bool deadDrawPassed
```

Field Value

[bool](#)

X

Position sur l'axe X de l'alien

```
public int x
```

Field Value

[int](#)

y

Position sur l'axe Y de l'alien

```
public int y
```

Field Value

[int](#)

Methods

AlienTouched(MissilePlayer)

check si un des missile du joueur à toucher un alien

```
public void AlienTouched(MissilePlayer missilePlayerTouched)
```

Parameters

missilePlayerTouched [MissilePlayer](#)

le missile du joueur

ChargementAlien(MissileAlien)

Charge un missile dans l'alien

```
public void ChargementAlien(MissileAlien missileAlien)
```

Parameters

missileAlien [MissileAlien](#)

Le missile charger provenant de la lsite des missiles

MisilleLaunchAlien(MissileAlien)

Fait en sorte que l'alien lance un missile

```
public void MisilleLaunchAlien(MissileAlien missileAlien)
```

Parameters

missileAlien [MissileAlien](#)

le missile de l'alien qui est dans la liste des missiles se trouvant sur lui

MoveRight()

Bouge à droite

```
public void MoveRight()
```

moveLeft()

Bouge à gauche

```
public void moveLeft()
```

Class Missile

Namespace: [Model](#)

Assembly: Model.dll

Classe contenant les caractéristiques principales des missiles du jeu ainsi que leur déplacement par défaut

```
public class Missile
```








Inheritance

[object](#)  ← Missile

Derived

[MissileAlien](#), [MissilePlayer](#)

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

Missile()

```
public Missile()
```

Fields

missileLaunched

```
public bool missileLaunched
```

Field Value

[bool](#) 

missileTouched

```
public bool missileTouched
```

Field Value

[bool](#)

X

```
public int x
```

Field Value

[int](#)

y

```
public int y
```

Field Value

[int](#)

Methods

MissileUpdate()

Update de la position du missile en le faisant aller vers le haut

```
public virtual void MissileUpdate()
```

Class MissileAlien

Namespace: [Model](#)

Assembly: Model.dll








Classe contenant les information et le déplacement des missiles envoyés par les aliens

```
public class MissileAlien : Missile
```

Inheritance

[object](#)  ← [Missile](#) ← MissileAlien

Inherited Members

[Missile.x](#), [Missile.y](#), [Missile.missileTouched](#), [Missile.missileLaunched](#), [object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) , [object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

MissileAlien(Alien)

```
public MissileAlien(Alien alien)
```

Parameters

alien [Alien](#)

Fields

SCREENHEIGHT

```
public const int SCREENHEIGHT = 40
```

Field Value

[int](#) 

Methods

MissileUpdate()

Nouvelle forme de MissileUpdate qui envoie le missile vers le bas

```
public override void MissileUpdate()
```


Class MissilePlayer

Namespace: [Model](#)

Assembly: Model.dll








Classe contenant les information et le déplacement des missiles envoyés par le joueur

```
public class MissilePlayer : Missile
```

Inheritance

[object](#)  ← [Missile](#) ← MissilePlayer

Inherited Members

[Missile.x](#) , [Missile.y](#) , [Missile.missileTouched](#) , [Missile.missileLaunched](#) , [Missile.MissileUpdate\(\)](#) , [object.Equals\(object\)](#)  , [object.Equals\(object, object\)](#)  , [object.GetHashCode\(\)](#)  , [object.GetType\(\)](#)  , [object.MemberwiseClone\(\)](#)  , [object.ReferenceEquals\(object, object\)](#)  , [object.ToString\(\)](#) 

Constructors

MissilePlayer(Player)

```
public MissilePlayer(Player player)
```

Parameters

player [Player](#)

Class Player

Namespace: [Model](#)

Assembly: Model.dll








Classe contenant toutes les informations sur le joueur ainsi que les actions qu'il peut réaliser

```
public class Player
```

Inheritance

[object](#)  ← Player

Inherited Members

[object.Equals\(object\)](#) , [object.Equals\(object, object\)](#) , [object.GetHashCode\(\)](#) , [object.GetType\(\)](#) ,
[object.MemberwiseClone\(\)](#) , [object.ReferenceEquals\(object, object\)](#) , [object.ToString\(\)](#) 

Constructors

Player(int, int)

```
public Player(int playerX, int playerY)
```

Parameters

playerX [int](#) 

playerY [int](#) 

Fields

bord

```
public bool bord
```

Field Value

[bool](#) 

playerDead

```
public bool playerDead
```

Field Value

[bool](#)

X

```
public int x
```

Field Value

[int](#)

y

```
public int y
```

Field Value

[int](#)

Methods

Chargement(MissilePlayer)

charge les missile dans le joueur

```
public void Chargement(MissilePlayer missile)
```

Parameters

missile [MissilePlayer](#)

missile du joueur

MisilleLaunch(MissilePlayer)

Lance le premier missile se trouvant dans la liste des missiles du joueur

```
public void MisilleLaunch(MissilePlayer missile)
```

Parameters

missile [MissilePlayer](#)

missile du joueur

PlayerTouched(MissileAlien)

Vérifie si le joueur est touché et tue le joueur si tel est le cas

```
public void PlayerTouched(MissileAlien missileAlienTouched)
```

Parameters

missilleAlienTouched [MissileAlien](#)

position du missile enemy

moveDown()

Bouge le joueur en bas

```
public void moveDown()
```

moveLeft()

Bouge le joueur à gauche

```
public void moveLeft()
```

moveRight()

Bouge le joueur à droite

```
public void moveRight()
```

moveUP()

Bouge le joueur en haut

```
public void moveUP()
```